

Worcester Polytechnic Institute Digital WPI

Masters Theses (All Theses, All Years)

Electronic Theses and Dissertations

2006-05-02

Tools to help build models that predict student learning

Ruta Sunil Upalekar

Worcester Polytechnic Institute

Follow this and additional works at: <https://digitalcommons.wpi.edu/etd-theses>

Repository Citation

Upalekar, Ruta Sunil, "Tools to help build models that predict student learning" (2006). *Masters Theses (All Theses, All Years)*. 634.
<https://digitalcommons.wpi.edu/etd-theses/634>

This thesis is brought to you for free and open access by [Digital WPI](#). It has been accepted for inclusion in Masters Theses (All Theses, All Years) by an authorized administrator of Digital WPI. For more information, please contact wpi-etd@wpi.edu.

Tools to help build models that predict student learning

by

Ruta S. Upalekar

A Thesis

Submitted to the Faculty

of the

WORCESTER POLYTECHNIC INSTITUTE

in partial fulfillment of the requirements for the

Degree of Master of Science

in

Computer Science

Date: May 2006

APPROVED:

Professor Neil Heffernan, Thesis Advisor

Professor David Brown, Thesis Reader

Professor Michael Gennert, Head of Department

Abstract

Analyzing human learning and performance accurately is one of the main goals of an Intelligent Tutoring System. The “ASSISTment¹” system [12] is a web-based system that blends assisting students and assessing their performance by providing feedback to the teachers. Good cognitive models are needed for an Intelligent Tutoring system to do a better job at predicting student performance. The ASSISTment system uses a method of cognitive modeling which is called a transfer model. A *Transfer Model* is a matrix that maps questions to skills. Other researchers have shown that transfer models help in building better predictive models that in-turn help in assessing a student’s performance [1, 8]. They provide a viable means of representing a subject matter expert’s view of which skills are needed to solve a given problem. However, the process of building a transfer model requires a lot of time. Reducing the time in which a transfer model is built would in turn help reduce the cost of building an Intelligent Tutoring System. Being able to build better transfer models will provide more efficient means of predicting learning in an intelligent tutoring system [6]. In this thesis we studied the creation of one transfer model that maps approximately the 263 released MCAS² items to approximately 90 skills. Recently, [5] and [9], using two different modeling methodologies, have both concluded that this transfer model can be used to predict MCAS scores more accurately. Currently the time spent in creating and storing a model is estimated to be approximately 65 hours. This thesis was motivated by the need of a set of tools that would reduce the time spent in building a transfer model. The goal of this thesis was to create a tool that

¹ The term “Assistment” was coined by Kenneth Koedinger and blends Assessment and Assisting.

² MCAS (Massachusetts Comprehensive Assessment System) is a graduation requirement in which all students educated with public funds in the tested grades are required to participate.

would speed up the process of building a transfer model. The efficiency of this tool is measured by an estimate of the overall time reduced for building a model. The average time reduced by using the tool on per question basis is also measured. The tool is not evaluated for its usability or for the ability to build better fitting transfer models.

Acknowledgements

I would like to thank my thesis advisor, Professor Neil Heffernan. He helped me with constant guidance and motivation to progress. My thesis reader, Professor David Brown, he gave me valuable feedback on the tool. Mrs. Christina Heffernan and Prof. Heffernan were the subject matter experts. Their knowledge and experience helped in creating the transfer models. I would also like to thank the ASSISTment team: Leena, Ming, Abe, Shane, Ema, Ryan, Emily, Derek, Andrei, and Hattis, who helped in the creation of the transfer models. Additionally I would like to acknowledge the US Department of Education, National Science Foundation, and Office of Naval Research, for providing the funding to accomplish this research. And last but not the least without the support of my loving husband I don't think this thesis would have been possible.

Table of Contents

1. INTRODUCTION	9
1.1 Motivation.....	10
2 BACKGROUND.....	11
2.1 Knowledge Components and Transfer Models.....	11
2.2 Towards finer grained Models	13
2.3 Transfer Model Uses: What good is a transfer model?.....	15
2.4 Building a transfer model.....	17
2.3 The Problem.....	21
3 FUNCTIONAL REQUIREMENT SPECIFICATION	22
4. IMPLEMENTATION	24
4.1 Features Implemented.....	25
4.1.1 Control Panel	25
4.1.1.1 Tree View of Questions and Knowledge Components.....	26
4.1.2 Graph Panel.....	29
4.1.3 Representation of Questions and Knowledge Components.....	30
4.1.4 Drag and Drop.....	33
4.1.5 Transfer Model as a Graph	34
4.1.5.1 Nodes and Edges.....	34
4.1.6 View	36
4.1.6.1 View Control.....	36
4.1.6.2 Zoom Control.....	37

4.1.7 Menu Options.....	38
4.1.7.1 File Menu	38
4.1.7.1.1 Open Transfer Model.....	38
4.1.7.1.2 Save Transfer Model.....	39
4.1.7.1.3 Print.....	39
4.1.7.2 AutoLayout (Menu)	40
4.1.7.2.1 Random Layout.....	40
4.1.7.3 Tools	41
4.1.7.3.1 Select all edges and nodes.....	41
4.1.7.3.2 Delete selected edges and nodes	41
4.1.7.3.3 Group Control	41
4.1.7.3.3.1 Creating, destroying and un-grouping a group	42
5. EVALUATION.....	43
5.1 Transfer Model Event I.....	43
5.2 Transfer Model Event II.....	46
5.3 Summary of the transfer model events	48
5.4 Transfer Model using the tool.....	49
5.5 Evaluation Summary.....	53
6. FUTURE ENHANCEMENTS.....	57
7. CONCLUSION.....	58
REFERENCES	59

Table of figures

Figure 1: Question 19 from the 2003 MCAS.....	12
Figure 2: A small piece of the WPI300 transfer model showing both how 14 questions (out of 245 in the WPI300) tap 19 knowledge components (out of 174 in the WPI300).....	14
Figure 3: Class summary report.....	16
Figure 4: On-going transfer model	17
Figure 5: Question cards and KC's (post-it notes).....	18
Figure 6: Skills entered using a web-based interface into the database.....	19
Figure 7: An item from the ASSISTment program called “Isosceles triangle”.....	19
Figure 8: A question tagged with a knowledge component.....	20
Figure 9: Control Panel.....	25
Figure 10: Tree Structure and KC description.....	26
Figure 11: Arrows used to maximize panels.....	27
Figure 12: Maximized view of the control panel.....	28
Figure 13: Graph Panel	29
Figure 14: Question and KC representation.....	30
Figure 15: Detailed description of a question	32
Figure 16: Nodes and Edges in a graph	35
Figure 17: View and Zoom Control.....	36
Figure 18: Graphical representation of a transfer model	39
Figure 19: Random Layout	40
Figure 20: Group Control.....	41
Figure 21: Group creation.....	42
Figure 22: KC's placed horizontally and questions placed randomly at the start of the evaluation.....	50
Figure 23: Mapping between KC's and questions at the end of evaluation.....	51

Table of Tables

Table 1 Approx. time taken for paper cut-outs for the 8th grade model	44
Table 2 Approx. time taken to enter the data in the database for the 8th grade model	45
Table 3 Approx. time taken for paper cut-outs for the 10th grade model	46
Table 4 Approx. time taken to enter the data in the database for the 10th grade model ..	47
Table 5 Time summary for 8th and 10th grade models	48
Table 6 Iterations in the mapping process using the tool.....	52
Table 7 Total time taken to build transfer models used in evaluation	53
Table 8 Average time spent per question for the 10th grade model	54

1. Introduction

Diagnostic score reports that can provide the outcome of the instructions in an assessment system is one of the most requested and an important feature of an assessment system. The ASSISTment system balances limited classroom time by integrating assisting with assessing. The system provides live reporting [3] to the teachers about the students' performance while the students are provided with assistance to solve a problem. Transfer Models are quite helpful for quality reporting [3]. A Transfer Model is a cognitive model that helps in predicting students' performance. It contains a group of knowledge components (KC's)³ and maps existing questions (original and scaffolding⁴) to one or more knowledge components. Transfer Models are also a good predictive measure of student learning as they are useful in selecting which can be the next best problem for a student based on his/her performance. Different mathematical modeling methodologies can be used to take a transfer model and make predictions including mixed-effects modeling [4], logistic regression [1, 5, 6] and Bayesian networks [9].

So, for an intelligent tutoring system to do a better job at assessing student performance, it is necessary that there is a good underlying transfer model. However, the process of building a transfer model is not an easy task. It involves a subject-matter expert creating knowledge components and mapping them to the existing questions. There is a lot of time and effort involved in creating a model and this is directly proportional to the number of questions that are to be mapped to the skills required to

³ The term knowledge component (KC) is used to describe skill(s) required to get a question right.

⁴ We use the term scaffolding question because they are like scaffolding that will help students solve the problem (and can "faded" later) so the scaffolds are meant to scaffold their learning.

solve them. The current method of building a transfer model is a manual process, where paper cut-outs of questions and skills are used. The mapping between questions and skills is done by placing the cut-out of a skill on a question that requires that skill. As the number of questions and skills increases, the space required for the mapping also increases and it becomes harder to represent the relation between the questions and the skills. The time taken for the process also increases as the number of questions increase. This thesis proposes a tool that will help reduce the time required to build transfer models. The tool will also provide a way to view a transfer model as a graph, in which the nodes are questions and skills. The mapping between the questions and skills will be represented by a directed edge. Currently, transfer models can be viewed as a two dimensional matrix that maps questions to skills.

1.1 Motivation

There is a lot of time spent in building a transfer model. Currently the time taken to build a transfer model is approximately 65 hours. Motivation for this thesis was driven by the demand to have tools to reduce the overall time taken to build a transfer model. The tool will also allow domain experts to view a transfer model as a graph, giving them a better idea of the relationship between the questions and the skills, than looking at the mapping in a matrix. The tool will also allow editing an existing transfer model.

2 Background

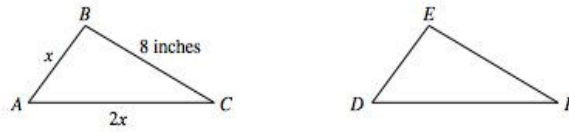
We will first understand what a transfer model is comprised of, before we see how one is built.

2.1 Knowledge Components and Transfer Models

A transfer model is a cognitive model that contains a group of knowledge components and maps existing questions (original items and scaffolding questions) to one, or more of the knowledge components [1]. It is called a transfer model since we hope to use the model to predict when learning and knowledge transfer will happen. In the psychometric community transfer models are called Q-Matrices [13, 14].

Knowledge Components have several names such as skills, strands, learning standards, etc. Knowledge Components may be described as a piece of acquired knowledge. Examples of knowledge components may be: concept, principle, production rule, schema, reasoning process, etc. *Massachusetts Curriculum Frameworks* breaks the 5 strands (Patterns, Relations and Algebra; Geometry; Data Analysis, Statistics and Probability; Measurement; Number Sense and Operations) into 39 “learning standards” for 8th grade math and tags each question with one of the 39 standards [11].

- 19 Triangles ABC and DEF shown below are congruent.



The perimeter of $\triangle ABC$ is 23 inches. What is the length of side \overline{DF} in $\triangle DEF$?

Figure 1: Question 19 from the 2003 MCAS

Let us look at an actual question, question 19 from the 2003 MCAS (Figure 1). In-order to get this question right, the student requires the understanding of concepts like congruence (Geometry), perimeter (Measurement), Algebra, and arithmetic operations (Number Sense). But the state classifies this question only under Geometry. Since the teachers are told to rely on the MCAS reports to plan their instruction, the student who gets the above problem wrong, will get instruction on Geometry. But the real problem might be Measurement, Algebra or Number Sense. So in order to give a detailed and an accurate report on the student's performance on problems, the ASSISTment Project aims at a finer grained transfer model.

2.2 Towards finer grained Models

Several attempts have been made at WPI towards finer grained transfer models. The first attempt was the transfer model called WPI300. The WPI300 contains around 174 knowledge components. Currently out of the 174 knowledge components, 102 are mapped to 853 questions. An instance of the WPI300 model is shown in Figure 2. The skills that are required to solve a problem are called Knowledge Components (KC) and the prerequisite skills are termed as Prerequisite Knowledge Components. The prerequisite knowledge components form a hierarchy in the transfer model. In this thesis we will not say much more about using hierarchies of skills as shown in Figure 2, but recently [8] had tried to investigate to see if using hierarchies of skills they can do a better job of predicting state test scores, but there is no clear result as of yet.

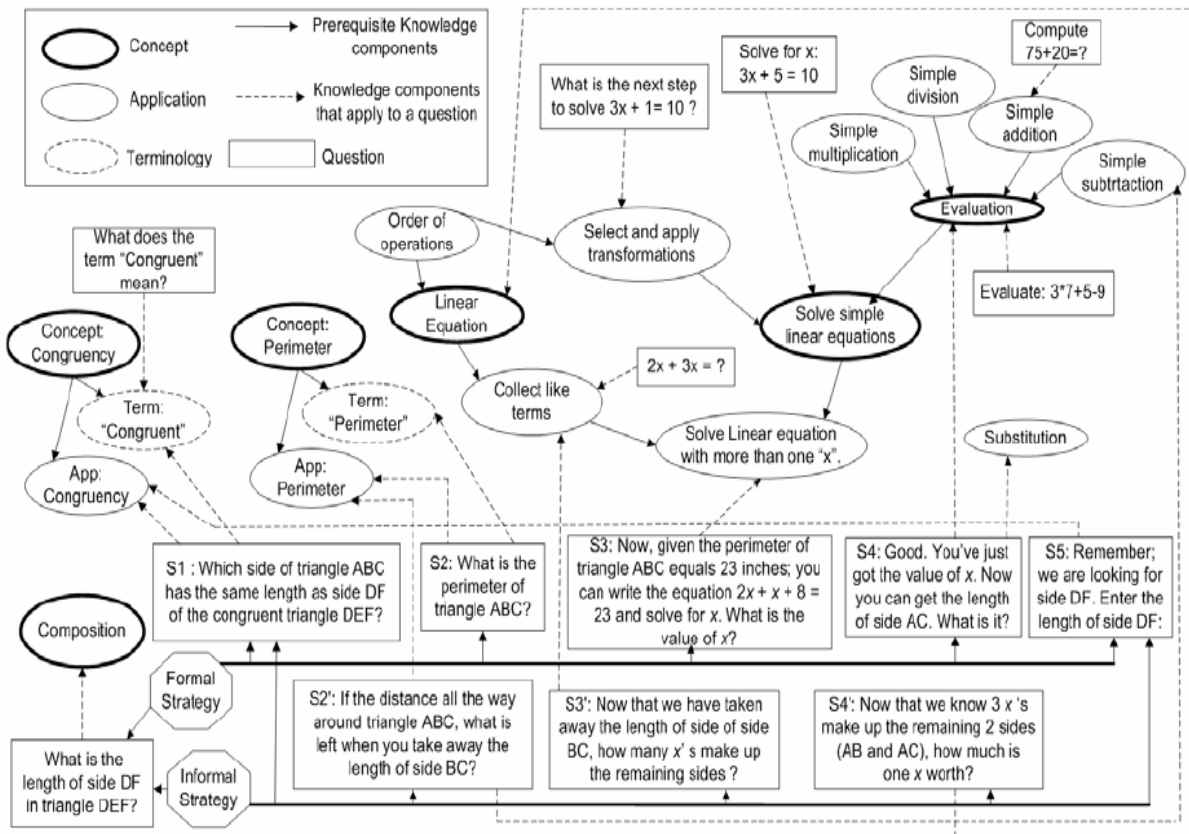


Figure 2: A small piece of the WPI300 transfer model showing both how 14 questions (out of 245 in the WPI300) tap 19 knowledge components (out of 174 in the WPI300).

Two recent attempts were made towards coding 8th Grade and 10th Grade MCAS questions (original questions and scaffolds). The 8th Grade model is called WPI-Apr-2005 and this was created in spring'05 and the 10th Grade model is called 10thGrade_WPI_Version1. Recently, the WPI-April-2005 model has been used to better predict students state test scores [5, 8, 9] which speaks to the value of engaging in such activity. We will look at the details of these two models a little later.

2.3 Transfer Model Uses: What good is a transfer model?

If we have a better transfer model, then we should be able to do a better job of predicting in real-time which items a student will get correct. Transfer Models help in selecting the next best problem for a student to work on. Transfer Models are also useful in better reporting. A “data-driven” approach is used in teaching where the teachers rely heavily on the reports provided to them to plan their teaching; hence providing accurate reports will help the teachers to plan better [4]. The ASSISTment Project provides “live reporting” to teachers. For example, one of the options provided to the teachers in reporting is a class summary report. This informs the teachers about the knowledge status of classes [3]. Teachers can select a transfer model and specify the number of knowledge components to be shown in the report. Figure 3 shows an instance of a class summary report. In this case the selected transfer model is “WPI-CMU-174-v1.0”. This shows the top 5 knowledge components that students in a particular teacher’s class are good at. Similarly the top 5 knowledge components that require more practice are also shown. Thus the teacher can obtain a better understanding about his/her class status regarding the knowledge components.

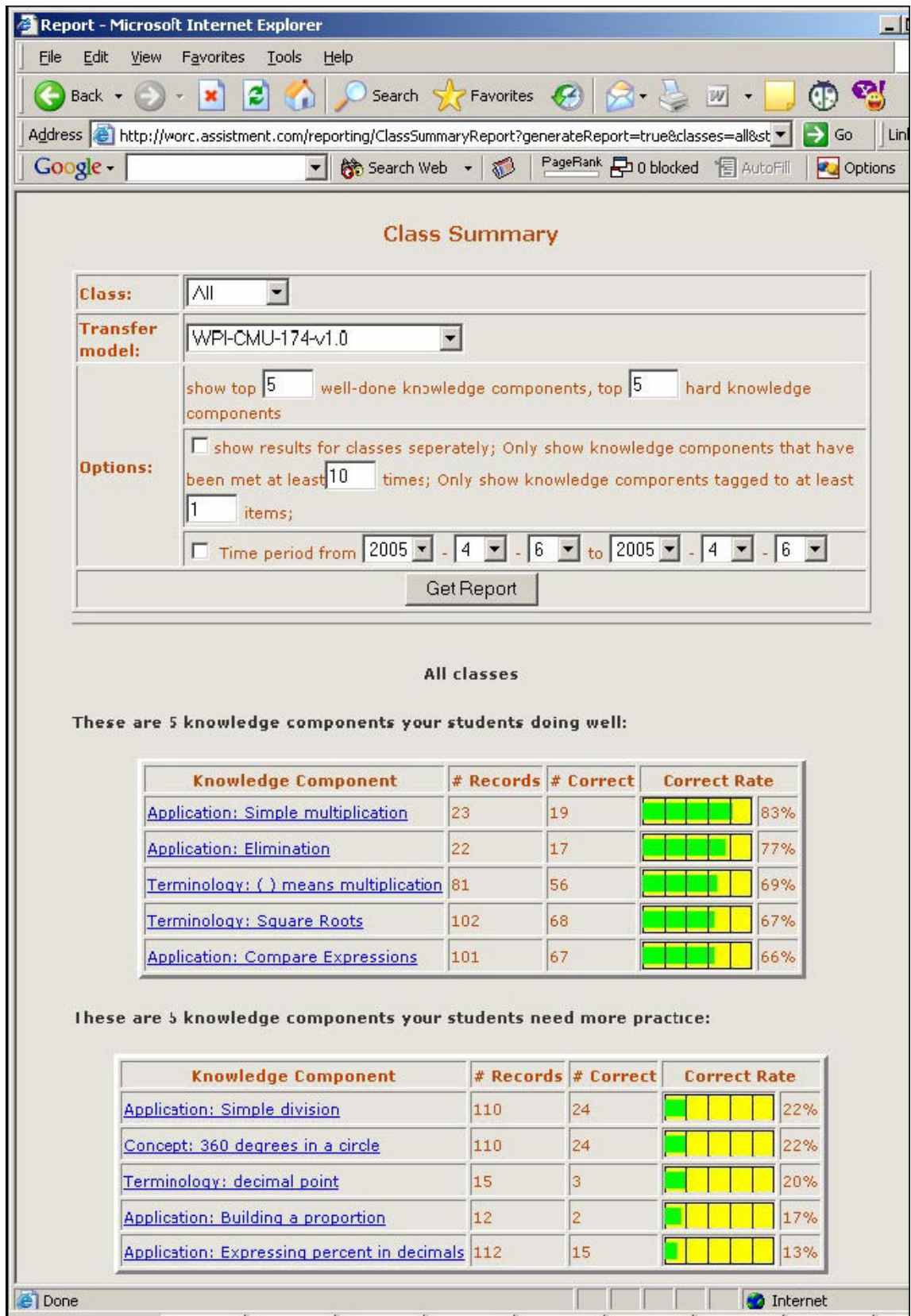


Figure 3: Class summary report

2.4 Building a transfer model

Currently transfer models are built by a subject matter expert. The questions are laid out as paper cut-outs on a large table. These are tagged with knowledge components by the subject matter expert. Questions that require a common knowledge component are stacked together. In this way, piles of questions tagged with knowledge components are created. Figure 4 shows a picture of an on-going transfer model building event. The questions are laid out as cards on tables and KC's are in the form of post-it notes.



Figure 4: On-going transfer model

Question-cards requiring a common KC are grouped together and labeled with a KC in the form of a post-it note. Once all the questions are tagged with the knowledge

components, the piles are put into folders. In this way each KC is put into one folder along with the questions that are tagged with that KC. Figure 5 shows question cards and KC's.

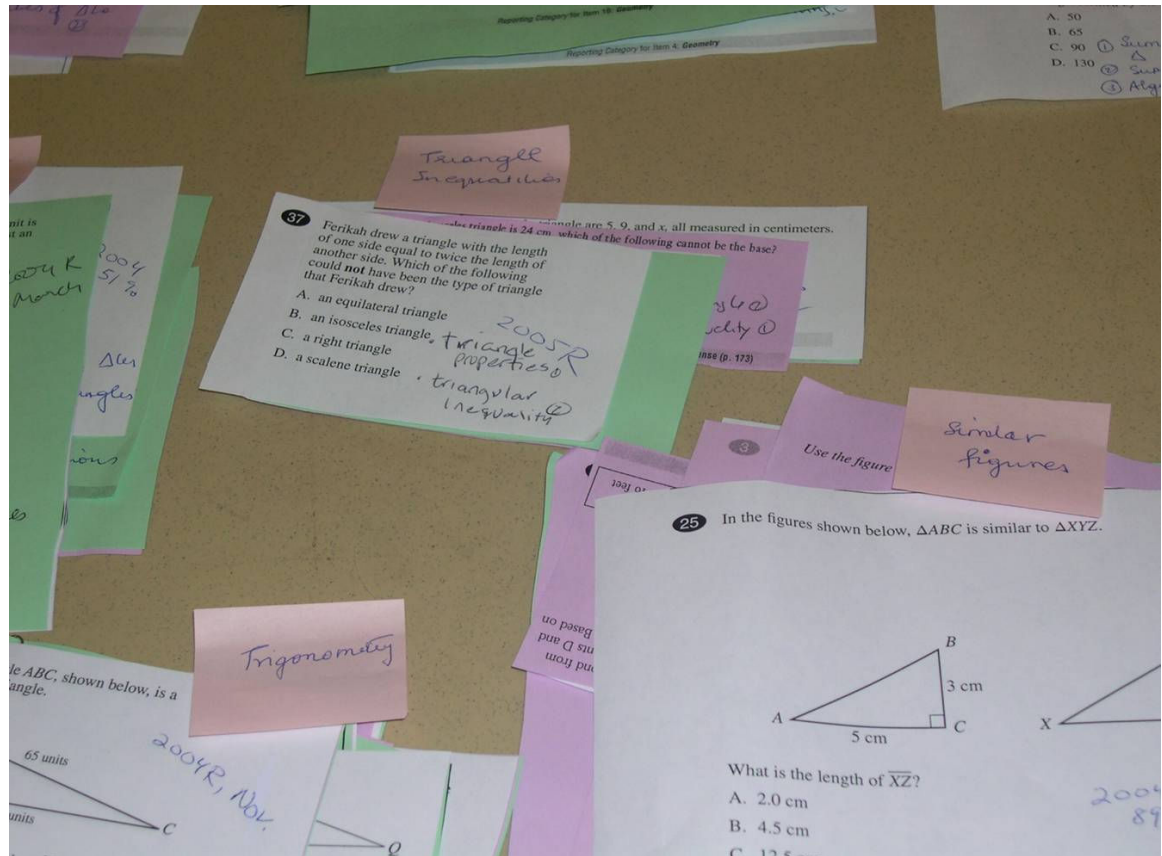


Figure 5: Question cards and KC's (post-it notes)

The transfer model that is created in this way has to be entered into a digital format to be stored in the ASSISTments database. This process is done in two steps. The first step is to enter the knowledge components into the database using a web-based interface. Figure 6 shows the interface used to enter the knowledge components into the database. To enter knowledge components, the user selects a transfer model from the drop-down list. Then he/she can enter the new knowledge component in the text field provided. The “Add the new knowledge component” button to add the knowledge

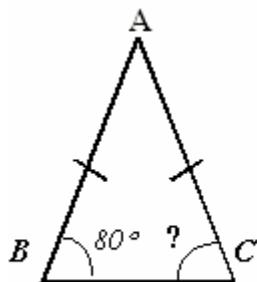
component to the transfer model. In this way each of the folders that contain KC's are looked up, and the corresponding KC's are added to the transfer model.

The screenshot shows a web-based interface with the following elements:

- Transfer models:** A dropdown menu currently showing "WPI-Apr-2005".
- Knowledge Component:** A dropdown menu currently showing "Venn-Diagram".
- Please type in the name of the new knowledge component:** An empty text input field.
- Add the new knowledge component:** A button located below the text input field.
- Please select your base transfer model:** A dropdown menu currently showing "WPI-Apr-2005".
- Import from selected transfer model:** A button located below the second dropdown menu.

Figure 6: Skills entered using a web-based interface into the database

The second step is to map the questions to the knowledge components in the transfer model. This is done by using a web-based interface called the “ASSISTment Builder” [15]. Let us see how an item is tagged using the builder. Figure 7 shows that an item⁵ by the name “Isosceles Triangle”.



In the figure shown above, triangle ABC is isosceles. What is the measure of angle ACB?

Figure 7: An item from the ASSISTment program called “Isosceles triangle”

⁵ We also call a question an item. An item contains the original questions along with scaffolding questions.

Figure 8 shows this item tagged with a knowledge component called “Isosceles-Triangle” from the transfer model WPI-Apr-2005. Once the knowledge component is selected, the user can then click the “Map to item” button to tag the question with the knowledge component.

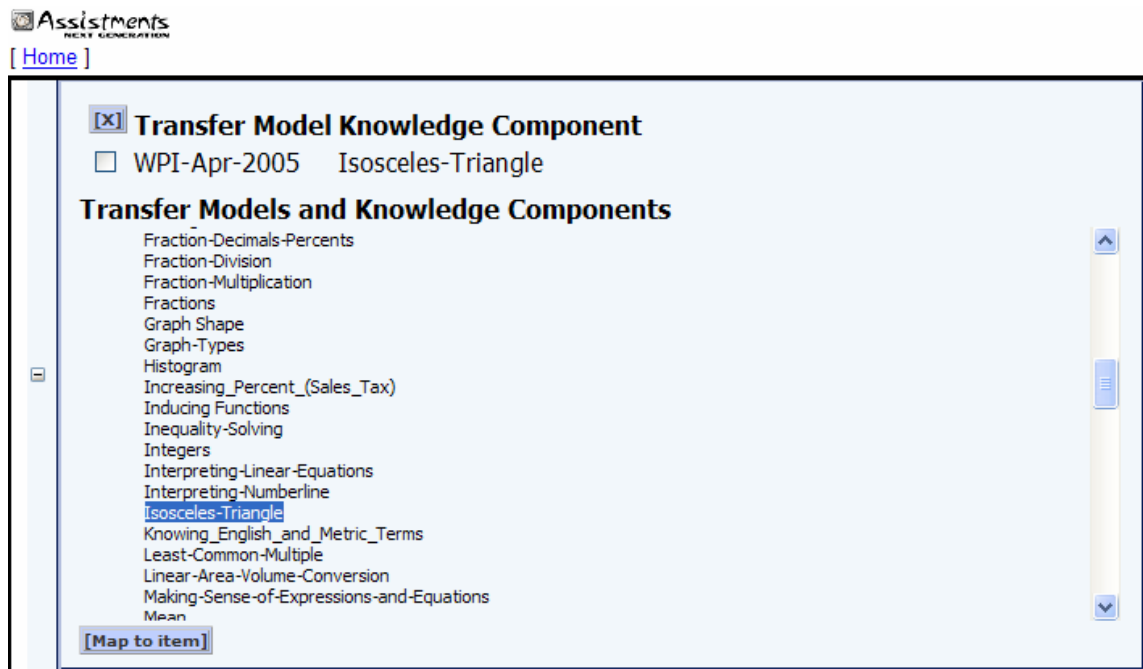


Figure 8: A question tagged with a knowledge component

2.3 The Problem

The process described above becomes difficult as the number of questions that are to be tagged increases. A number of reasons can be attributed to this increase in complexity. The time taken for the process increases as the number of questions increase. As the number of questions increases, a larger surface-area may be required for the mapping process. One question may have multiple skills required and so copies of that question have to be made. As the number of questions increase, managing the question cards becomes difficult. The layout of the cards also has to be managed because the transfer model mapping depends on the physical arrangement of the cards. Errors might occur because of displacement of the cards and the probability of an error increases as the number of questions increase. The folders that hold the cards also have to be carefully stored until the transfer model is entered into the database. The cards may be misplaced from the folders and a wrong representation of a transfer model may be entered into the database. Also, the transfer model that is built cannot be viewed as a graph, so normal users such as teachers will not be able to easily view an existing model. Currently a transfer model can be viewed as a two dimensional matrix. As the number of questions increase, the mapping between questions and KC's becomes difficult to locate. Viewing the transfer model as a graph will might make it easier to see the mapping between the questions and the KC's. There was a need for a tool that would reduce the time taken in the process of building a transfer model. This tool would also help in viewing a transfer model and modifying an existing transfer model.

3 Functional Requirement Specification

Developing the tool involved gathering the requirement specifications. The first step was gathering the requirement specifications. The requirement specifications were as follows:

1. Layout: This included the general appearance of the tool and the representation of the transfer model elements.
 - i. A graph panel to create a graph.
 - ii. Nodes are knowledge components and questions.
 - iii. The edges between the knowledge components and questions define the required knowledge components.
 - iv. Questions and knowledge components represented as cards.
 - v. The cards must be dragged to the canvas and relationship between the skills and questions must be an edge.
 - vi. Grouping of similar cards. Questions that have similar skills must be grouped together.
 - vii. The user should be able to focus on the cards that are laid out on the canvas, a zoom-in/ zoom-out feature should be available.
2. Saving:
 - i. The graph that is created must be saved as a transfer model representation.
 - ii. A picture of the graph for viewing must also be available locally.
3. Editing and Viewing:
 - i. Users should be able to view existing transfer models.
 - ii. Users should also be able to edit transfer models.

4. Printing :

- i. Users should be able to print the graphical representation of the model.

4. Implementation

The tool was implemented in Java [7]. A graph drawing library called Visualizing Graphs with Java (VGJ) [16] was used in the development. Basic guidelines for developing a graphical user interface [10] were also followed. Eclipse [2] was used as the Integrated Development Environment for the development of the tool.

The tool was implemented according to the requirement specifications. At every stage of development, the tool was tested for the features implemented and additional features were added if required.

4.1 Features Implemented

The following features were implemented in the tool. Every feature is explained in detail with a screenshot of that feature.

4.1.1 Control Panel

Figure 9 shows the view of the tool. The control panel is the left side of the tool where the KC's and the questions are displayed. The controls for scaling the image are located on the left-bottom side of the tool.

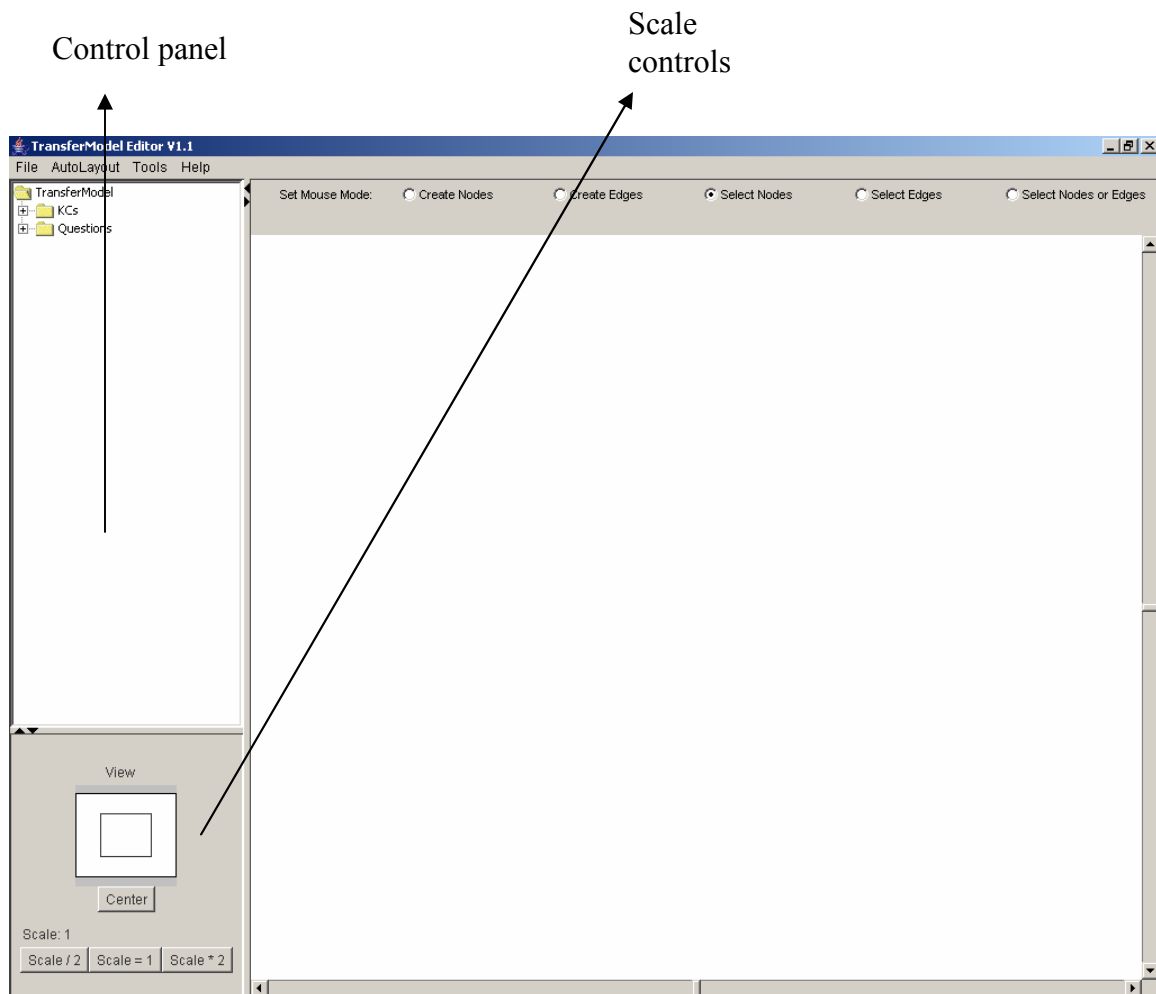


Figure 9: Control Panel

4.1.1.1 Tree View of Questions and Knowledge Components

The questions and the KC's are displayed in the control section in a tree format. Figure 10 shows an expanded KC. To make it easier for the user to navigate, a pop-up menu is added. The description of a KC will appear when the cursor is moved over that KC. In Figure 10, the KC “Area-of-Circle is shown selected. This occurs when the cursor is moved over that KC using a mouse. Using this feature the user can get a preview of the description without having to expand the folders.

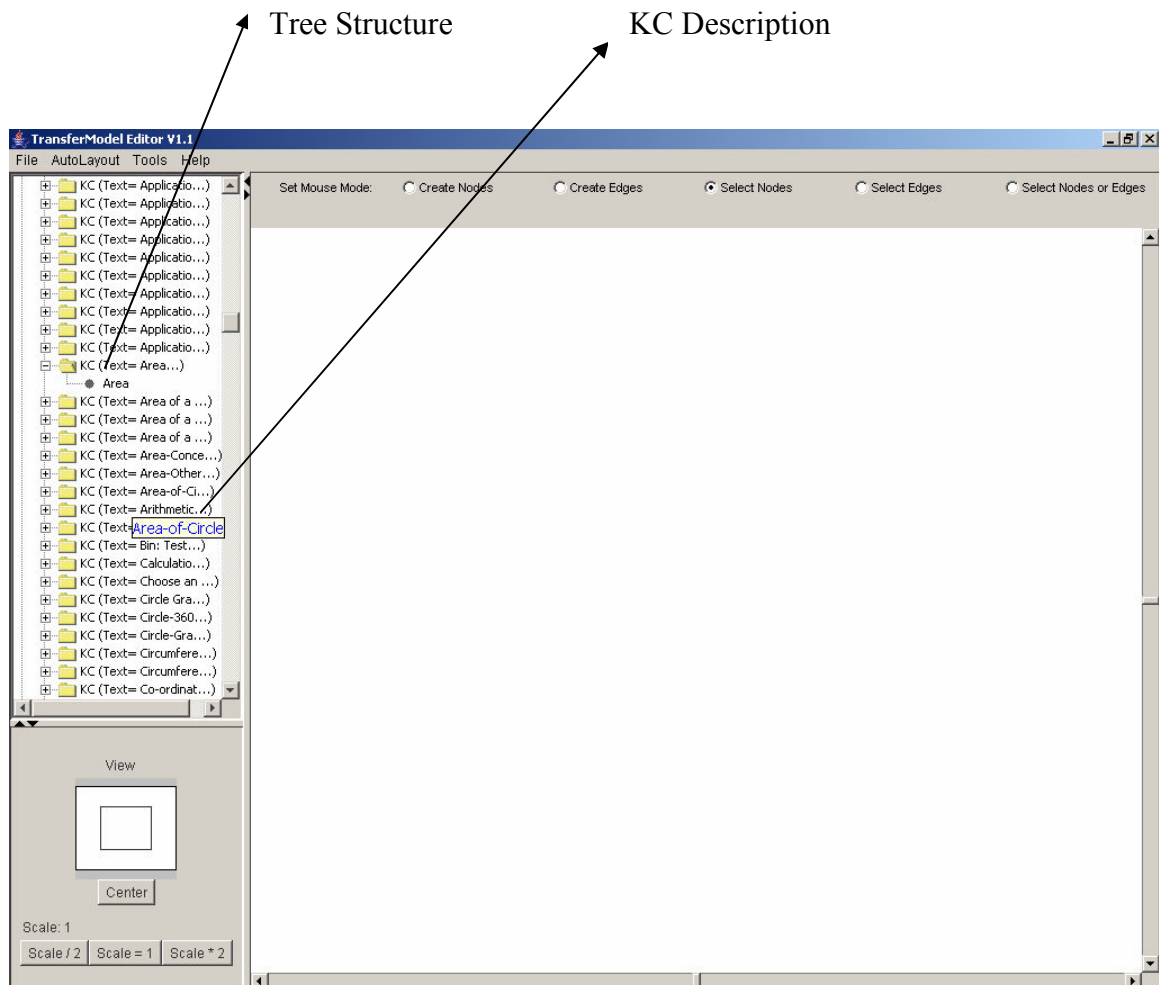


Figure 10: Tree Structure and KC description

The control panel can also be maximized by clicking on one of the arrows as shown in Figure 11. This allows the user to read through the KC's and questions without any space restriction. Clicking on the arrow again will restore the original view. Figure 12 shows the control panel when it is expanded.

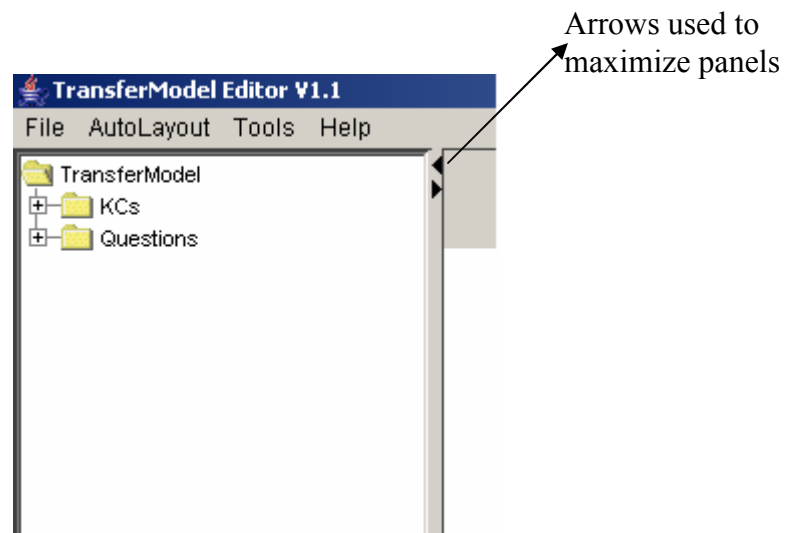


Figure 11: Arrows used to maximize panels

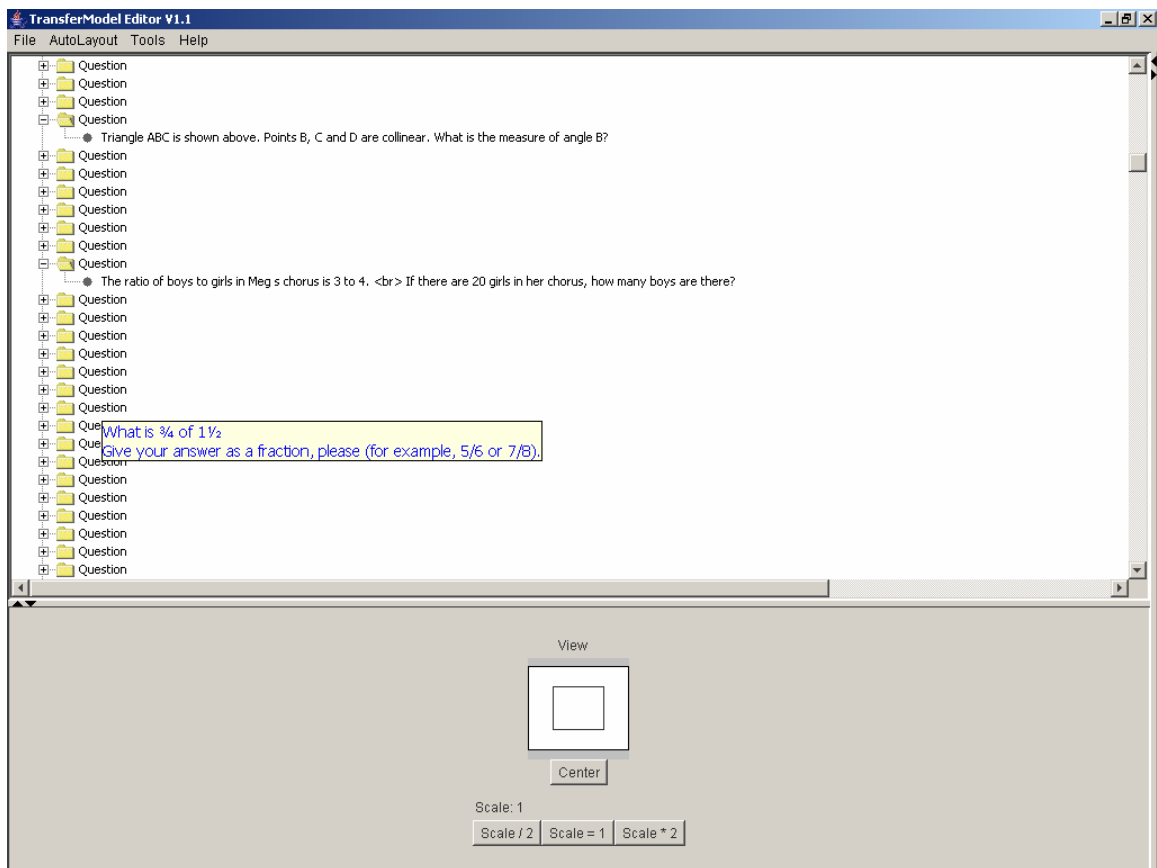


Figure 12: Maximized view of the control panel

4.1.2 Graph Panel

The Graph panel occupies the right side of the tool window. It contains a graph canvas. The graph canvas displays a transfer model as a graph. The panel can be used to create a graphical representation of a transfer model or to view an existing transfer model as a graph. Figure 13 shows the graph panel in the tool. The radio buttons on top of the graph panel are used to set the mouse mode. The different modes available are “Create Nodes”, “Create Edges”, “Select Nodes”, “Select Edges” and “Select Nodes or Edges”. These modes are set by selecting the radio buttons. The selected model will be reflected in the graph panel.

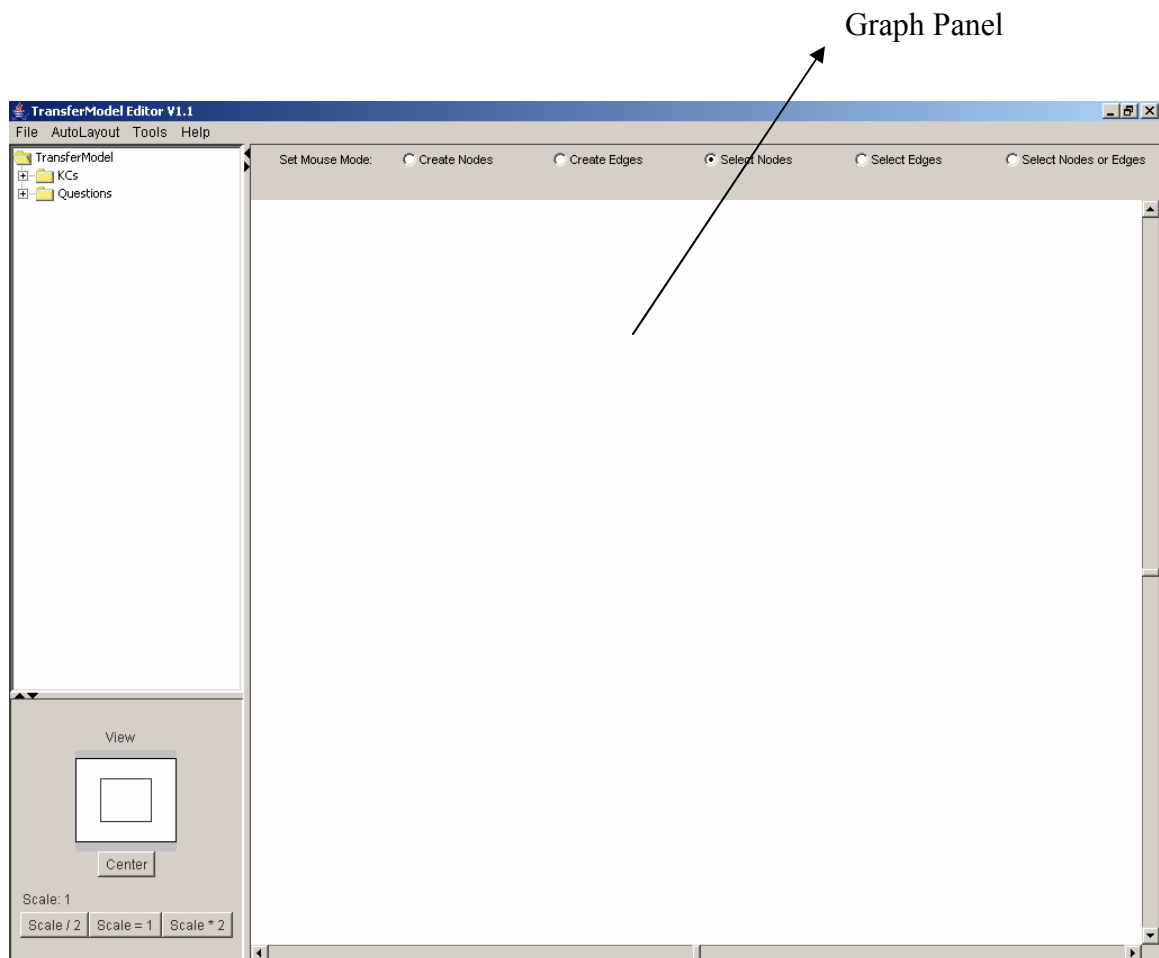


Figure 13: Graph Panel

4.1.3 Representation of Questions and Knowledge Components

In the graphical representation of the transfer model, questions are represented as rectangles and knowledge components as ovals. Questions that require a knowledge component as a pre-requisite are connected to that knowledge component by a directed arrow. The direction of the edge is always from the question to the KC. On an attempt to draw the edge from the KC to the question a beep is sounded to indicate error. Figure 14 is a screenshot that shows a question connected to a knowledge component called “Multiply and Divide”.

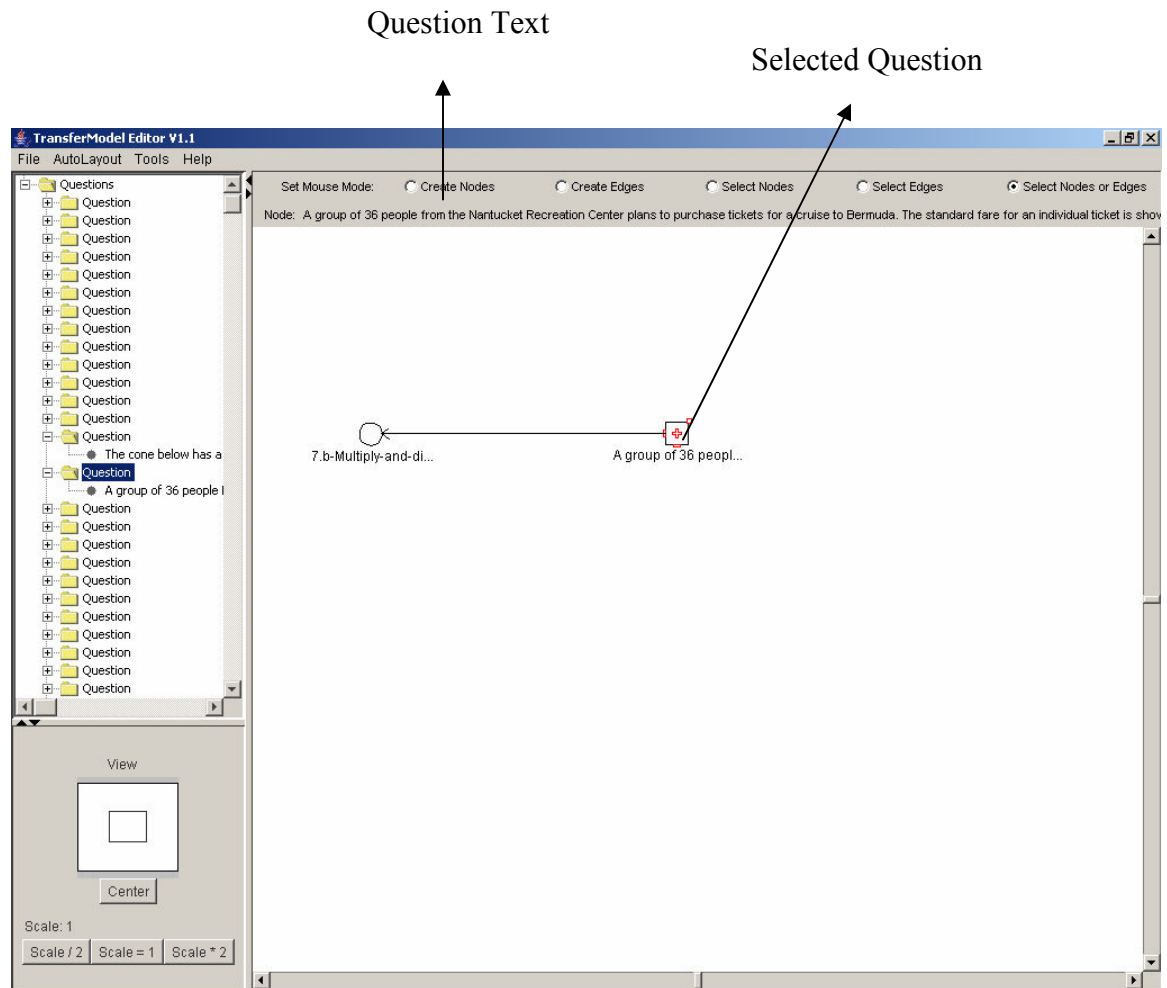


Figure 14: Question and KC representation

The graph panel also displays the text of questions. This is shown in Figure 14. This is displayed at the top of the panel when that question node is selected. The text of a KC is also displayed on the graph panel when that KC node is selected. The “+” mark on the question node in Figure 14 indicates that the question node is selected.

The text of a question or a knowledge component in the graph panel is read by double clicking on a question or a knowledge component node. Figure 15 is a screenshot of what is displayed when a node is double clicked on the graph panel. A window with the details of that node is displayed. The position of the node on the graph panel is displayed with the x and y co-ordinates of the node. The z co-ordinate is always zero. The size of the node is also displayed specifying the height, width and the depth of the node. The shape of the node is also shown. The KC node will be oval and the question node will be rectangle. Figure 15 shows the details for a question node and hence the shape is a rectangle. The position of the label with respect to the node is also shown. The position can be either below the node or above the node. The image for a node can also be displayed. Currently, images in nodes are not supported and so image details are not present. The text of the node is displayed in the text box labeled data. This ensures that as the number of questions increase on the graph panel, users can still read details for every node for better understanding. This feature allows the user to read the question and map the knowledge component(s) required based on the question. It is important to be able to read the complete text of a question in order to decide which knowledge component is required to get that question right.

Node 1

Position:

X Y

Bounding Box:

Height Width Depth

Shape:

Label:

Label Position:

Image: (Leave Height and Width blank for automatic sizing.)

Type:

Source:

Data:

Figure 15: Detailed description of a question

Question Text

4.1.4 Drag and Drop

Question and knowledge components from the control panel on the left side of the tool can be dragged and dropped on the graph panel. When a question is dragged and dropped it is represented as a rectangle: a knowledge component is represented as an oval. This allows the user to conveniently select the questions and knowledge components from the control panel and use them in building a transfer model.

4.1.5 Transfer Model as a Graph

The following sections describe how the transfer model is viewed as a graph.

4.1.5.1 Nodes and Edges

In the graphical representation of the transfer model, questions and knowledge components form the nodes and the relation between them forms a directed edge. The edge always starts from a question to a knowledge component. Nodes can be either created or dragged from the existing questions and knowledge components from the control panel. Figure 16 shows the panel that contains controls for setting the mouse mode. The modes are selecting and creating a node or an edge. Multiple nodes and edges can be selected by clicking on the “Select Nodes or Edges” on the mouse mode panel. The control can be toggled by selecting the radio buttons in the panel. If a node or an edge has to be deleted the node has to be selected first and then deleted. Deleting a node or an edge can be done either by hitting the “Delete” key on the keyboard or by going to the “Tools” menu and selecting “Delete Selected Items”.

Panel for setting mouse

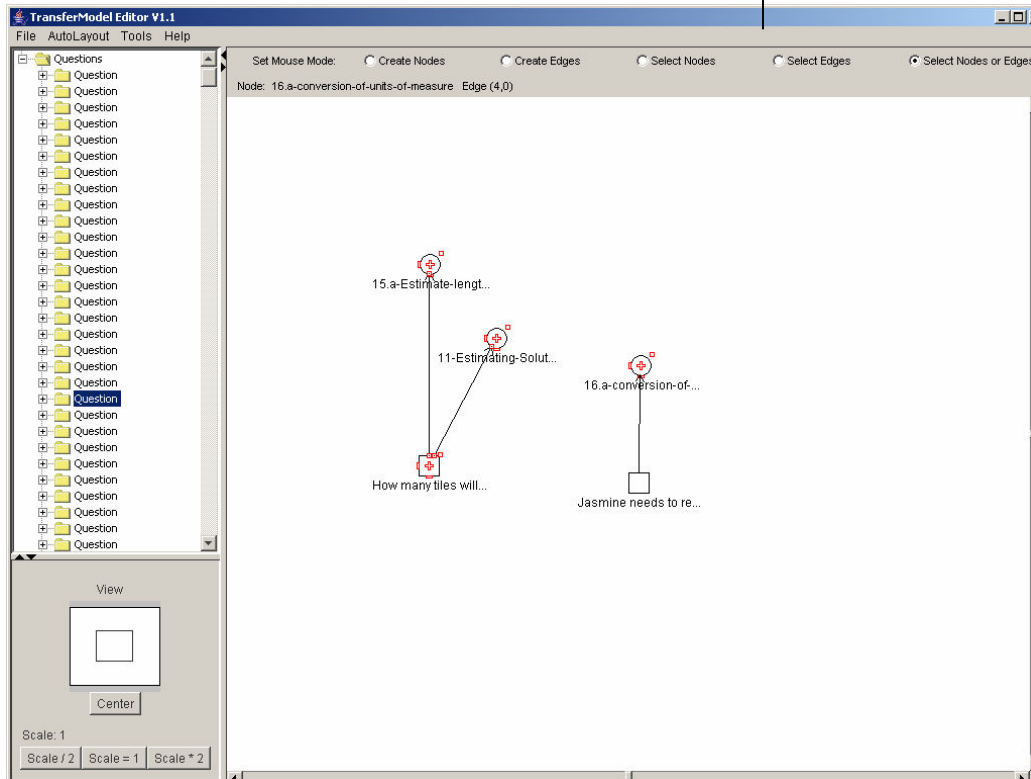


Figure 16: Nodes and Edges in a graph

4.1.6 View

The view section of the tool contains the view control and the zoom control.

4.1.6.1 View Control

The tool has a view control feature. The view control is located in the control panel at the bottom. Figure 17 shows the section of the tool that contains the View control. The area of the outer rectangle in Figure 17 represents the area of the graph canvas and the inner rectangle represents what is on the canvas. The view control can be used to move the transfer model to desired place on the graph canvas. This minimizes the need to select all the nodes and edges in the transfer model required to move the graph. This is an additional feature that helps in the layout of the graph. The view is controlled by moving the small rectangle and is relative to the area of the graph canvas, which is the outer rectangle. The “center” button will place what is on the graph canvas in the center of the canvas.

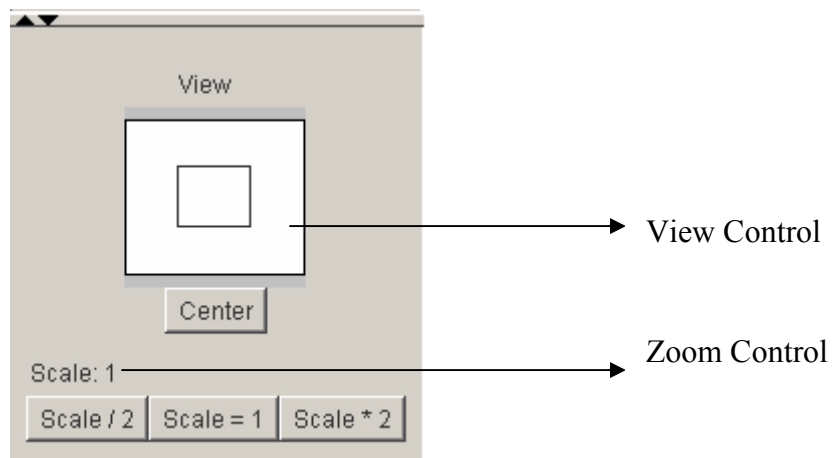


Figure 17: View and Zoom Control

4.1.6.2 Zoom Control

The graph can be scaled up and down using the zoom control. Figure 17 shows the zoom control. The original scale value is 1. The current scale of the graph is also displayed as “Scale = 1” in Figure 17. The scale is decreased or increased using the “Scale / 2” and “Scale * 2” buttons. The scale is increased or decreased based on the current value of the scale. The original size can be restored by clicking on the button named “Scale =1”. This feature helps to see the entire graph layout by zooming out if the graph is very big. A general idea of the transfer model can be achieved by viewing the entire graph. For example certain knowledge components have a lot of questions attached to them. The domain expert can get a general idea about those knowledge components by looking at the area of the graph that has most number of edges.

4.1.7 Menu Options

The following sections describe the menu options available in the editor.

4.1.7.1 File Menu

The file menu and its sub-sections are described below.

4.1.7.1.1 Open Transfer Model

This menu option is for opening an existing transfer model. Transfer models that are created and stored can be displayed in the tool for editing or viewing. The format of the transfer model is in a graphical modeling language (GML) format. Any changes done on the transfer model can be saved locally in the GML format. In order to obtain the transfer models from the database in the GML format, a database utility is run periodically and the desired transfer models from the database are stored locally in the GML format. They can then be opened in the tool and viewed as a graph. Figure 18 shows the default layout of the transfer model. The default layout enables the graph to be viewed clearly without the over-lapping of the nodes.

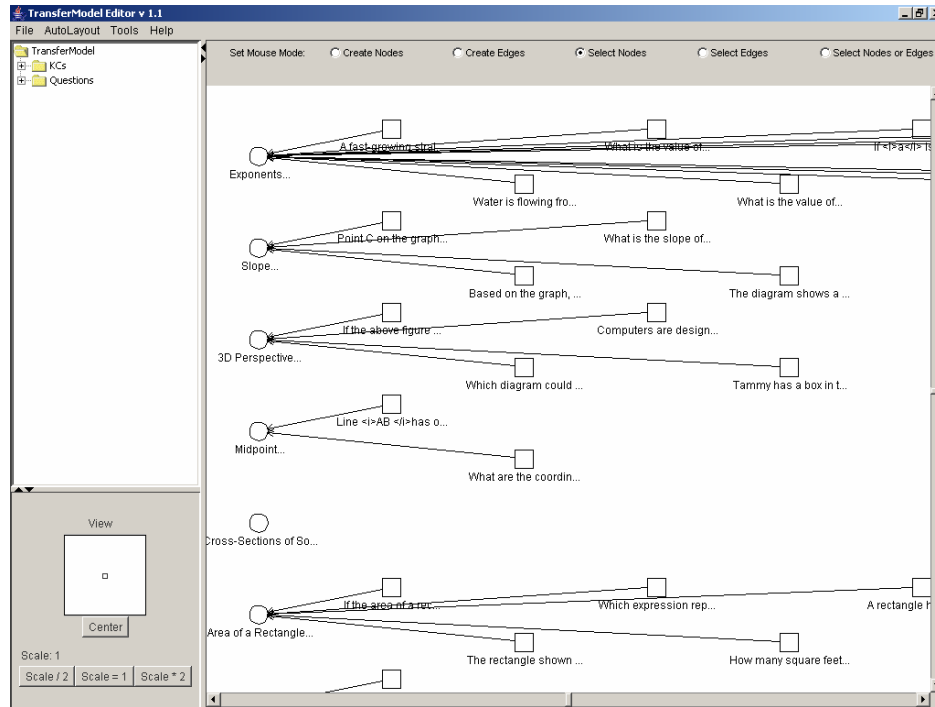


Figure 18: Graphical representation of a transfer model

4.1.7.1.2 Save Transfer Model

Transfer Models can be saved locally in the GML format. The connectivity of the transfer model and the formatting is saved. Models saved in the GML format can be opened again for modification and saved again.

4.1.7.1.3 Print

Transfer Models in the form of a graph can be printed in a PDF format. This helps in better understanding of the model. The model can be printed by specifying the desired dimensions for the print. Once the dimensions are specified, a file is saved in a PDF format and can be then printed.

4.1.7.2 AutoLayout (Menu)

The autolayout menu option is described below.

4.1.7.2.1 Random Layout

The automatic layout menu has the random layout option. Figure 19 shows a random layout applied to the default layout of a transfer model. This may be used if the user wishes to change the default layout of the graph. In the default layout, the knowledge components in a transfer model are laid out vertically and the questions connected to the knowledge components are laid out horizontally. This layout gives a better understanding of the relation of the knowledge components with the questions.

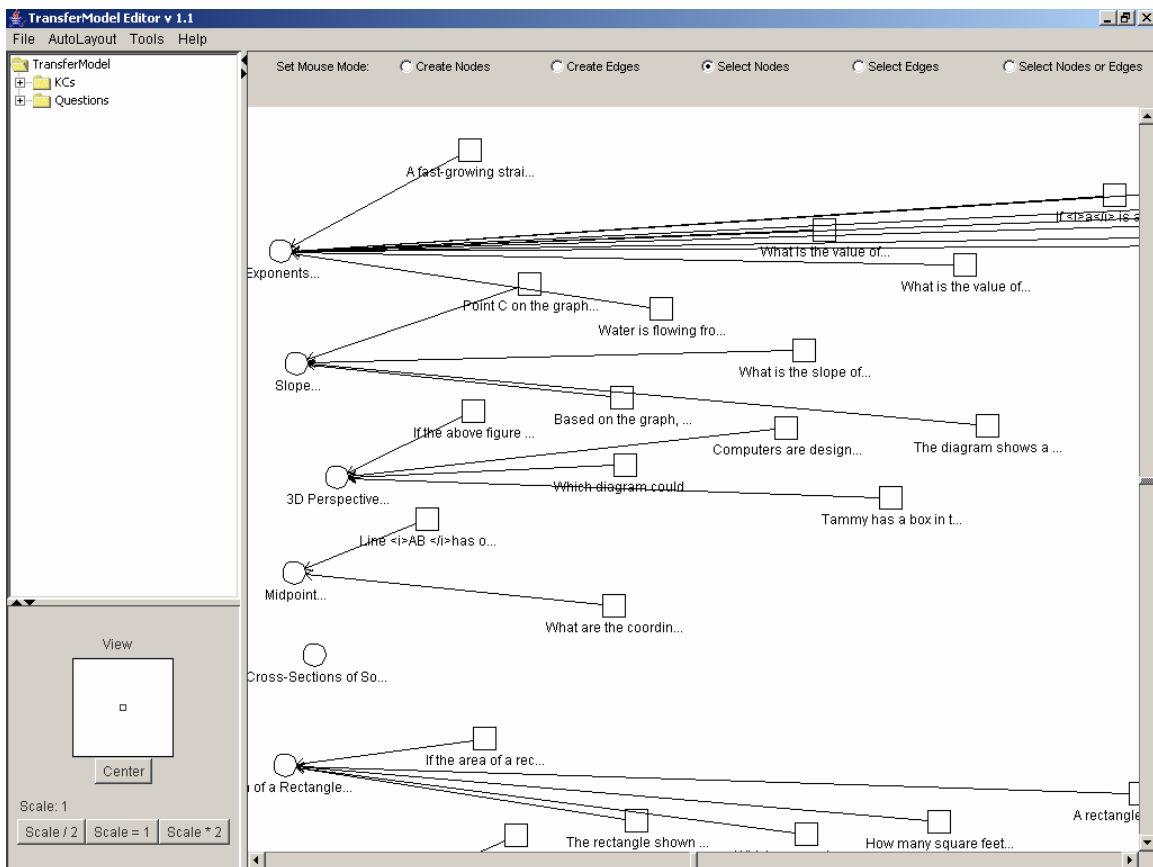


Figure 19: Random Layout

4.1.7.3 Tools

The tools menu has some features that are implemented such as group control, and options for managing edges and nodes. These features are described below.

4.1.7.3.1 *Select all edges and nodes*

All the edges and nodes in the graph panel can be selected by using the “Select All” option in the tools menu.

4.1.7.3.2 *Delete selected edges and nodes*

Selected edges and nodes can be deleted using this option from the tools menu. An alternate method to delete selected nodes and edges would be hitting the delete key on the keyboard.

4.1.7.3.3 *Group Control*

Group control feature helps to create a group of nodes and edges. This feature can be used to create a group, destroy a group and un-group a group. Figure 20 shows the various options that are available in the group control feature.

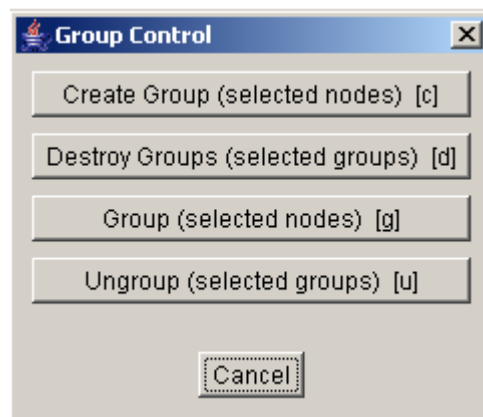


Figure 20: Group Control

4.1.7.3.3.1 Creating, destroying and un-grouping a group

Figure 21 shows a group that is created from selected nodes and edges. When a group is created a name is given by default. This name can be changed by double clicking on the group node. Using the group feature, a selected group can be deleted by using the “Destroy Groups” option. The “Ungroup” option can be used to un-group selected nodes and edges to their state before they were grouped.

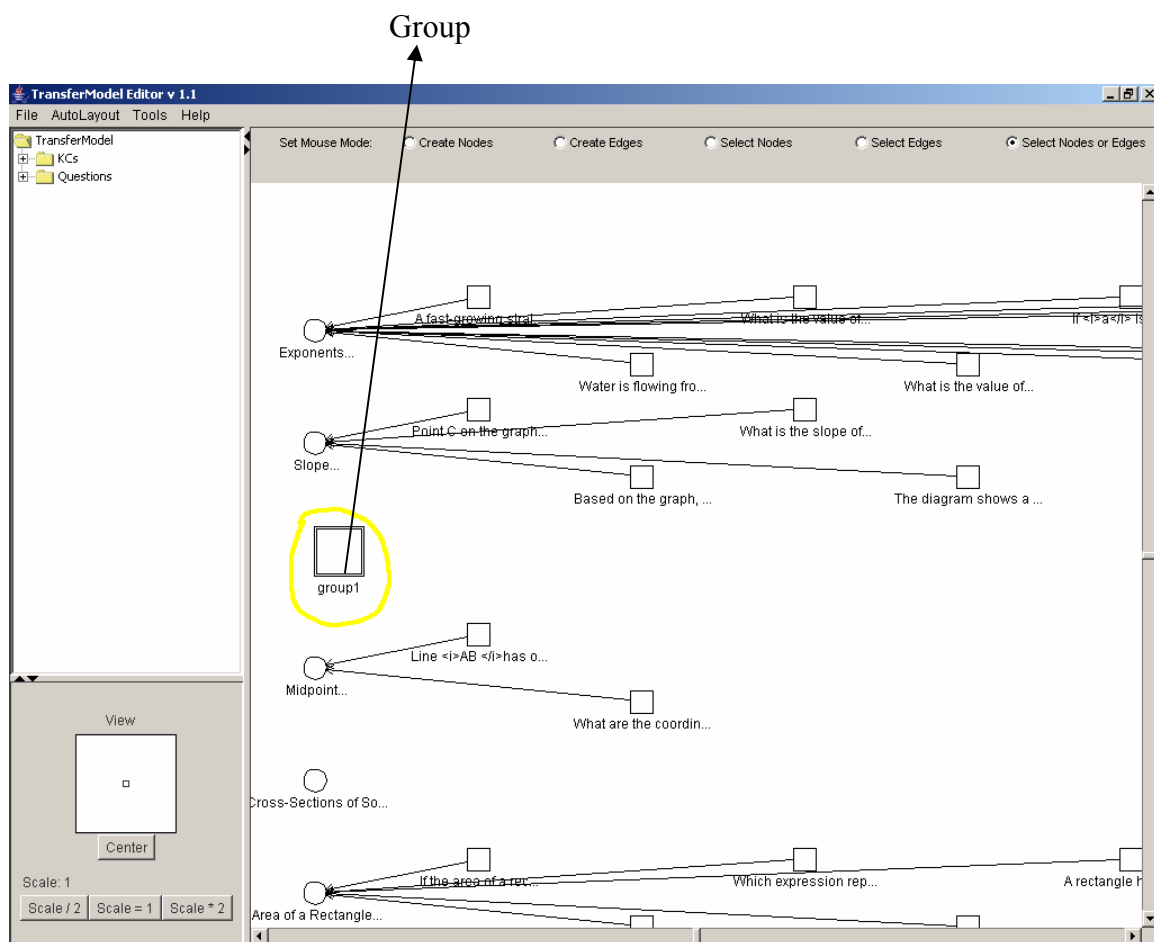


Figure 21: Group creation

5. Evaluation

Evaluation was based on the comparison of the time taken to build a transfer model without using the tool to the time taken to build a transfer model using the tool. In the following sections we describe the transfer models that were created without using the tool. Further we compare the time taken for the process using the tool with the time taken for the earlier transfer models.

5.1 Transfer Model Event I

The first transfer model event was for the 8th grade model event. For this transfer model, 8th grade items (original and scaffolding) from the ASSISTment database were used. There is no record of exactly how many total questions were used in building the transfer model, but we did start with the 263 released items that were available at that time, and we certainly tagged at least 95% of those items. Given, that we had already built many scaffolding questions for approximately half of the released items, and each items would get about 3 scaffolding questions on average, we tried to tag all the individual scaffolding questions with skills. We estimate that that we tagged 200-400 scaffolding questions at that time. In addition, we tagged approximately a few hundred more scaffolding questions, but the exact number of those is not known.

Before the actual building of the transfer model, paper cut-outs of the questions had to be done. This involved 5 people and the time taken by every person was recorded. Table 1 shows the approximate time taken for making paper cut-outs. The time taken was recorded based on the response from the people involved in the process. Some responses were a range of time taken and hence it is an approximate value. The upper limit of the time was taken when there was a range of time.

Name	Approx. time in hours
Shane	6 (4 - 6)
Abe	5 (4 - 5)
Ema	8
Ryan	3
Ruta	7
Total	29 (26 - 29)

Table 1: Approx. time taken for paper cut-outs for the 8th grade model

The main section of the event was the mapping of the questions to the knowledge components. The model was built by one subject-matter expert. The total time taken by the subject-matter expert to map questions to the knowledge components for this model was 6.5 hours. Once the mapping of the questions was done, the paper cut-outs were arranged in folders that contained questions mapped to knowledge components.

The next step was to enter the transfer model built into the database. This process was done using a web-interface. This was done by a total of 4 people. The table below shows the approximate time taken for this process.

Name	Approx. time in hours
Andrei	12 (8 – 12)
Derek	9
Hattis	12
Emily	12
Total	45 (41 – 45)

Table 2: Approx. time taken to enter the data in the database for the 8th grade model

5.2 Transfer Model Event II

The second event that was used in the evaluation of the tool was the 10th grade model. This model was also built in a way similar to the 8th grade model described above. MCAS test and retest questions for 10th grade from the year 1998 to 2005 with the exception of the open-response questions were used in this model. In contrast to the 8th grade model where original and scaffolding questions were used, only original questions were used in this model. The total number of questions used in this model was 464. This model was built by two subject-matter experts. The total time taken by the subject matter experts for the mapping process was 6.5 hours. The table below shows the approximate time taken to make the paper cut-outs for this event.

Name	Approx. time in hours
Leena	8
Ruta	7
Total	15

Table 3: Approx. time taken for paper cut-outs for the 10th grade model

The approximate time taken to enter the transfer model into the database is given in the table below.

Name	Approx. time in hours
Ming	8 (7 – 8)
Leena	5
Ruta	3
Abe	5
Total	21 (20 – 21)

Table 4: Approx. time taken to enter the data in the database for the 10th grade model

5.3 Summary of the transfer model events

The following table summarizes the approximate time taken for the two transfer model events described above.

Tasks	8thGrade Model (hours)	10th Grade Model (hours)
Paper cut-outs	29	18
Mapping Event	6.5	13
Enter KC's in DB	No Data	3
Enter Question-KC map to DB	21	45
	Total = 56.5	Total = 79

Table 5: Time summary for 8th and 10th grade models

5.4 Transfer Model using the tool

The evaluation was done by placing the 10th grade knowledge components horizontally at the top of the graph panel of the tool, and placing the 10th grade questions randomly on the panel. The expert had to map the questions to the knowledge components. Existing 10th grade knowledge components were used for the evaluation. The expert had familiarity with the placement of the knowledge components on the graph panel before the evaluation was done. Also, the expert was familiar with the use of the tool.

The evaluation compared the reduction in the time taken to build the model using the tool compared to the time taken to build the 10th grade model without using the tool. This evaluation is based only on the comparison of time factor before and after using the tool and ignores human computer interaction, security measures and user learn-ability. A total of 68 knowledge components were placed linearly on the graph panel and 324 10th grade questions were placed randomly. Figure 22 shows the graph panel at the start of the evaluation process.

Figure 23 shows a screen shot of the tool at the end of the evaluation. There were totally 8 iterations in which 118 questions were mapped. The total time taken was 3.06 hours. At the end of the last iteration, out of the 324 questions, 162 questions were remaining and 44 questions were attempted but could not be mapped.

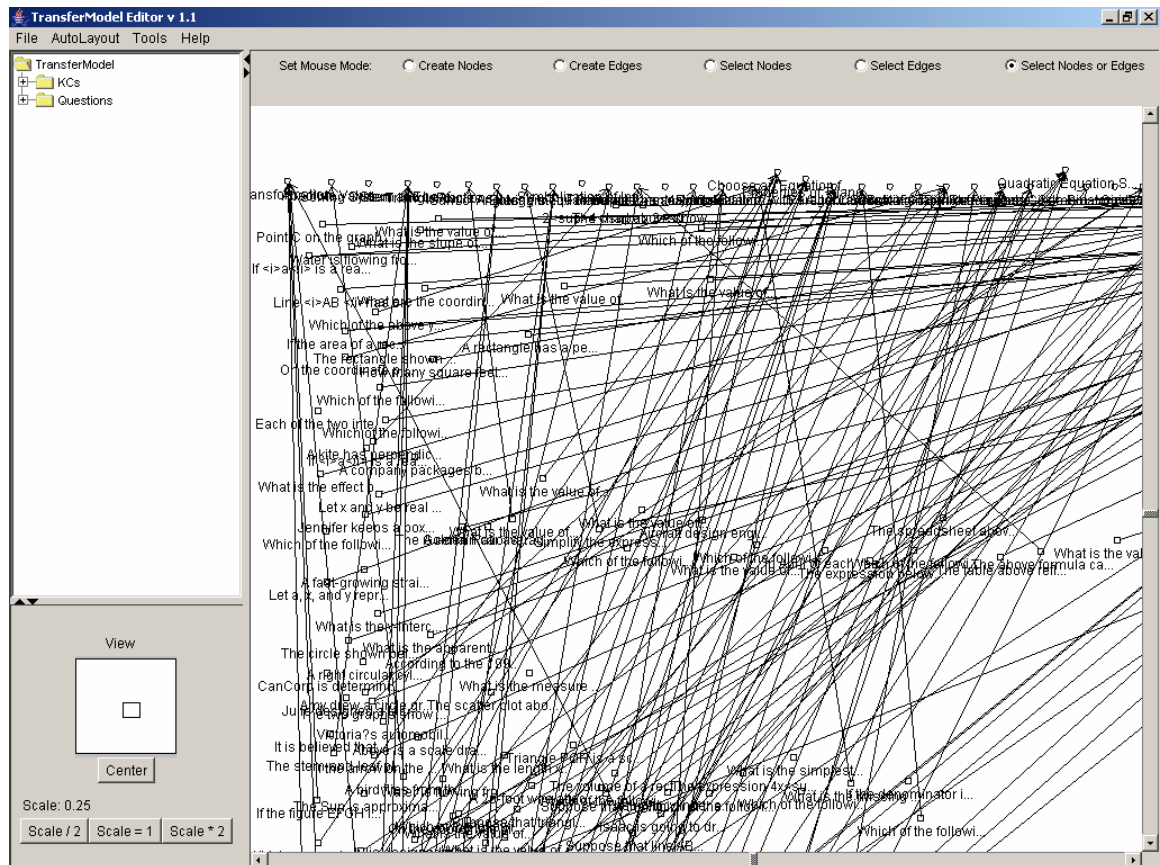


Figure 23: Mapping between KC's and questions at the end of evaluation

The process was carried out in iterations. In each iteration a few questions were mapped to the KC's. Every iteration was timed and the number of questions that were mapped was recorded. Table 6 shows all the iterations with the recorded information.

Iterations	Start Time	End Time	Total Time (End Time – Start Time in minutes)	No of questions marked	Avg. Time/question (in minutes)
1	7:43	8:12	64	29	2.21
2	8:26	8:51	25	17	1.47
3	9:10	9:17	14	07	2.00
4	9:20	9:36	16	14	1.14
5	9:43	9:51	08	10	0.80
6	10:00	10:15	15	11	1.36
7	12:01	12:13	12	08	1.50
8	12:30	1:00	30	22	1.36
			$\Sigma = 184$	$\Sigma = 118$	1.559

Table 6: Iterations in the mapping process using the tool

5.5 Evaluation Summary

Table 7 shows the total time taken to build the 10th grade model and the time taken to build the test model using the tool for the evaluation.

Tasks	10th Grade Model (hours)	Test Model built with Tool (hours)
Paper cut-outs	18	Did not do
Mapping Event	13	3.06
Enter KC's in DB	3	3
Enter Question-KC map to DB	45	Did not do
	Total = 79	Total = 6.06 *

Table 7: Total time taken to build transfer models used in evaluation

The test model built with the tool completely eliminates the need for paper cut-outs and does not require entering the data into the database. However, for the test model, we used the existing KC's from the 10th grade model and so the time taken to enter the KC's in the database is also accounted. We know that a total of 464 questions were used in the 10th grade model. We can hypothesize that if 118 questions were mapped in 6.06 hours using the tool, then 464 questions may be mapped in 23.83 hours. We can compare the estimated total time for the 10th grade model which is 79 hours to the estimated time

* This is based on certain assumptions, such as the expert's familiarity with the KC's before evaluation and also the familiarity with the use of the tool.

in the hypothesis which is 23.83 hours. Using the tool to build the transfer model has reduced the time taken to build the model to almost 1/3rd of the time taken to build the model without using the tool.

Table 8 shows the average time spent on per question basis for the 10th grade model without using the tool. The total number of questions used in the 10th grade model is 464. We see from the table that the average time spent per question is 0.168 hours, which is about 10 minutes. We can compare the average time taken per question for the model using the tool from table 6, which is 1.5 minutes. We can see that by using the tool the average time spent per question spent without using the tool was reduced by a factor of 6 after using the tool.

Tasks	10th Grade Model (hours)	Average time spent per question (hours)
Paper cut-outs	18	0.038
Mapping Event	13	0.028
Enter KC's in DB	3	0.006
Enter Question-KC map to DB	45	0.096
	Total = 79	Total = 0.168

Table 8: Average time spent per question for the 10th grade model

The tool has met with several requirements. It is now possible to create new knowledge components and questions faster using the tool compared to the current method. It is also possible to view the transfer model as a graph. Transfer models can also be edited, saved locally and printed. Viewing a transfer model is convenient with the

additional features like view and zoom control. The tool completely eliminates the need for paper cut-outs and makes it possible for researchers and teachers to build their own transfer models in a much lesser time than compared to the existing method.

6. Future Enhancements

The tool is functional when a transfer model is to be viewed as a graph, but there are enhancements that can be added to the tool. Currently there is no way of evaluating the transfer models that are created. Future work on the tool may involve making use of log-data of student performance on questions that are mapped according to the transfer model created by using the tool to evaluate the performance of that model. This feedback is very useful for researchers when they are trying to compare the performance of two transfer models for research purposes. Security measures were not included in this version of the tool. User access based security measures can also be added.

The questions that are displayed in the tool do not contain images that may be present in the questions, so adding the images to the questions is a feature enhancement that can be added in the future. Currently the tool does not focus on the human computer interaction aspect. The tool can be improved by making several graphical user interface design enhancements to make the interface more user-friendly.

7. Conclusion

The tool meets most of the functional requirements. It is now possible to view a transfer model as a graph. It is possible to edit an existing transfer model and also create one. The tool makes it possible for teachers and researchers to create their own transfer models. One of the main achievements of the tool is the reduction of the time it takes to build a model; the time taken to build a transfer model by using the tool is about 1/3rd of the time taken to build the model without using the tool. The time taken based on per question basis is also reduced by 6 times as shown in the evaluation. Thus the evaluations show that use of the tool could save time and resources. Adding advanced features will help users in better user learn-ability the tool. A good transfer model will mean better reports to teachers. This will also mean reduction in cost for the ASSISTment Project.

References

1. Croteau, E., Heffernan, N. T. & Koedinger, K. R. (2004) Why Are Algebra Word Problems Difficult? Using Tutorial Log Files and the Power Law of Learning to Select the Best Fitting Cognitive Model, *7th Annual Intelligent Tutoring Systems Conference*, Maceio, Brazil
2. Eclipse, a Java IDE <http://www.eclipse.org/>
3. Feng, M., Heffernan, N.T., (2005). Informing Teachers Live about Student Learning: Reporting in the Assistment System. Submitted to the *12th Annual Conference on Artificial Intelligence in Education*, Amsterdam.
4. Feng, M., Heffernan, N., & Koedinger, K. (2006) “Predicting state test scores better with intelligent tutoring systems: developing metrics to measure assistance required.” *8th Annual International Conference on Intelligent Tutoring System*, Taiwan.
5. Feng, M., Heffernan, N., Mani, M. & Heffernan, C. (in press) “Does using Finer Grained models lead to better predictions of state test scores” Appearing in the “*Educational Datamining Workshop*”, part of the *8th Annual International Conference on Intelligent Tutoring System*.
6. Freyberger, J., Heffernan, N., & Ruiz, C. (2004) Using Association Rules to Guide a Search for Best Fitting Transfer Models of Student Learning Workshop on “Analyzing Student-Tutor Interaction Logs to Improve Educational Outcomes” at the *Proceedings of 7th Annual Intelligent Tutoring Systems Conference*. Maceio, Brazil.
7. Java Homepage <http://java.sun.com/>

8. Pardos, Z., Heffernan, N., Anderson, B., & Heffernan, C. (in submission) "Using Skill Hierarchy and parameter learning to improve test prediction with Bayesian Networks." *Appearing in the "Educational Datamining Workshop", part of AAAI, 2006, Boston.*
9. Pardos, Z., Heffernan, N., Anderson, B., & Heffernan, C. (in press) "Using Fine-grained skill models to fit student performance with Bayesian Networks" *Appearing in the "Educational Datamining Workshop", part of the 8th Annual International Conference on Intelligent Tutoring System.*
10. Principles of Good GUI design
http://www.iie.org.mx/Monitor/v01n03/ar_ihc2.htm
11. Published by Massachusetts Department of Education (2005), Massachusetts Comprehensive Assessment System, <http://www.doe.mass.edu/mcas/>
12. Razzaq, L, Feng, M., Nuzzo-Jones, G., Heffernan, N.T. et. al (2005). The Assistment Project: Blending Assessment and Assisting. In C.K. Looi, G. McCalla, B. Bredeweg & J. Breuker (Eds.) *Proceedings of the 12th Artificial Intelligence In Education*, 555-562. Amsterdam: ISO Press
13. Tatsuoka, K. Kikumi Rule Space (1983): An approach for dealing with misconceptions based on item response theory. *Journal of Educational Measurement*, 20(4), 345-354.
14. Tiffany Barnes, The Q-matrix method: mining student response data for knowledge. *AAAI-2005 workshop on "Educational DataMining"*, Pittsburgh.
15. Turner T.E., Lourenco A.L.N., Heffernan N.T., Macasek M.A., Nuzzo-Jones G., (2006). The Assistment Builder: An Analysis of ITS Content Creation Lifecycle, *Submitted to FLAIRS2006*, 555-562. Florida, USA.

16. Visualizing graphs with Java (VGJ):

http://www.eng.auburn.edu/department/cse/research/graph_drawing/graph_drawing.html